## AMENDMENTS TO THE CLAIMS

The following listing of claims will replace all prior versions, and listings, of claims in the application.

1. (Currently Amended) A system for parallel asynchronous command execution, comprising:

a first computer system for directing a call to invoke a remote procedure in a second computer system, the first computer and second computer communicating *via* a non-persistent connection;

wherein the second computer system performs remote execution of the remote procedure and upon completion of the remote procedure generates an event trigger and transmits the event trigger and remote procedure results to the first computer system;

wherein the first computer system carries out other procedures while waiting for the event trigger and remote procedure results from the second computer system.

2. (Original) The system of claim 1 further comprising a distributed object architecture for communicating between the first computer system and the second computer system.

3. (Original) The system of claim 2 wherein the distributed object architecture is implemented *via* at least one of COM, DCOM, and CORBA interface languages.

4. (Original) The system of claim 1 wherein the first computer system configures an event to receive the remote procedure results from the second computer system.

5. (Original) The system of claim 4 wherein the event is a Windows Management Infrastructure event.

6. (Original) The system of claim 4 wherein the event is provided with an identifier for enabling the second computer system to notify the first computer system.

7. (Original) The system of claim 1 further comprising a work item and a thread for processing the remote procedure.

8. (Original) The system of claim 1 further comprising a completion event on the second computer system for notifying the first computer system.

9. (Original) The system of claim 1 further comprising an object interface for providing remote access between the first computer system and the second computer system.

10. (Original) The system of claim 9 wherein the object interface further includes an identification for a remote object.

11. (Original) The system of claim 9 wherein the object interface further includes a computer name for identifying where to trigger an event.

12. (Original) The system of claim 9 wherein the object interface further includes an identification for an event for the second computer system to trigger.

13. (Original) The system of claim 9 wherein the object interface further includes an input argument for providing results from the remote procedure.

14. (Original) The system of claim 9 wherein the object interface further includes at least one of a username, domain, and password for specifying a user context for the remote procedure.

15. (Original) The system of claim 1 wherein the second computer system further comprises a work interface for executing the remote procedure.

16. (Original) The system of claim 15 wherein the work interface further comprises an input argument and an output argument.

17. (Currently Amended) A system for parallel asynchronous command execution, comprising:

a first computer system for directing at least one call to invoke a remote procedure in at least one other computer system, the first computer system continuing to perform other tasks, the at least one other computer system upon completion of the remote procedure establishing a non-persistent connection to the first computer system, the at least one other computer system generating an event trigger and transmitting the event trigger and remote procedure results to the first computer system.

18. (Currently Amended) A method for parallel asynchronous command execution, comprising the steps of:

directing at least one call from a first computer system to invoke a remote procedure in at least one other computer system;

executing other functions with the first computer system;

establishing a non-persistent connection between the first computer system and the at least one other computer system upon completion by the at least one other computer system of the remote procedure; and

generating an event trigger and transmitting the event trigger and remote procedure results to the first computer system.

19. (Original) The method of claim 18 further comprising the step of communicating between the first computer system and the at least one other computer system *via* a distributed object architecture.

20. (Original) The method of claim 19 wherein the distributed object architecture is implemented *via* at least one of COM, DCOM, and CORBA interface languages.

21. (Original) The method of claim 18 further comprising the step of configuring an event on the first computer system to receive results from the at least one other computer system.

22. (Original) The method of claim 21 wherein the event is a Windows Management Infrastructure event.

23. (Currently Amended) A system for parallel asynchronous command execution, comprising:

means for directing at least one call from a first computer system to invoke a remote procedure in at least one other computer system;

means for establishing a non-persistent connection between the first computer system and the at least one other computer system upon completion by the at least one other computer system of the remote procedure; and

means for generating an event trigger and transmitting the event trigger and remote procedure results to the first computer system.

24. (Currently Amended) A system for parallel asynchronous command execution, comprising:

a server for performing and responding to at least one remote call by invoking a remote procedure;

the server establishes a non-persistent connection to communicate results of the remote procedure; and

the server generates an event trigger and transmits the event trigger and remote procedure results upon completion of the remote procedure.

25. (Original) The system of claim 24 wherein a client receives the event trigger and remote procedure results *via* the non-persistent connection.

26. (Original) The system of claim 25 wherein the client configures an event to receive the remote procedure results.

27. (Currently Amended) A system for parallel asynchronous command execution, comprising:

a first computer for directing a call to invoke a remote procedure in a second computer, the first computer transmits a non-persistent signal to the second computer;

wherein the second computer executes the remote procedure and upon completion of the remote procedure generates an event trigger and transmits the event trigger and remote procedure results to the first computer *via* the signal.

28. (Original) A computer-readable medium having computer-readable instructions for performing the acts of, comprising:

responding to at least one remote call by invoking a remote procedure;

establishing a non-persistent connection to communicate results of the remote procedure; and

generating an event trigger and transmitting the event trigger and remote procedure results upon completion of the remote procedure.

29. (Original) The computer-readable medium of claim 28 further comprising, receiving the event trigger and remote procedure results *via* the non-persistent connection.

30. (Original) The computer-readable medium of claim 28 further comprising, configuring an event to receive the remote procedure results.